Universitat Politècnica de Catalunya
Facultat d'Informàtica de Barcelona

Information Sharing and Filtering in Communities of Social Networks

Guillem Pérez Delgado

March 2018

Abstract

Social networks are very present in our everyday lives: we are very used to them even though we do not fully understand them. Although different social networks may seem to be very different to the naked eye, it has been observed that social networks show some common properties (e.g. power law degree distribution and small-world phenomenon) which leads us to believe that there are some underlying principles that all social networks obey that can explain these common properties. The goal of this project is to try to shed some light in the field of social networks trying to uncover some of these underlying principles. In particular, with this project we will try to answer the question of how information is shared and/or filtered in a community of a social network.

Contents

Chapter 1

Introduction

In this first chapter, the topic of research of the project will be presented as well as the motivation to study it and its objectives. Additionally, in the second section, an outline of the structure of the rest of the project will be presented in order to guide the reader throughout the project.

1.1 Motivation and Objectives

Over the last few years, social networking applications such as Facebook, Twitter and Instagram have shown an incredible growth in their use. These applications allow people and organizations to create user profiles and connect with each other i.e. these applications allow its users to build social networks.

Knowing this, it seems that social networks are very present in our everyday lives, so in this sense they are very important. Nevertheless, we do not fully understand social networks yet. For example, we do not have answer to simple questions such as *Why do we build social networks?* or *How do we use social networks?*.

One property of social networks is the existence of a community structure. Communities are groups of individuals that interact with one another and share some common characteristics.

Having a community structure in social networks provides a classification of the individuals in it that is beneficial to the individuals themselves because communities make it easier to interact with other individuals similar to them.

One of the most important facets of the communities of social networks (but not the only one) is the sharing of information: individuals in a community use it to share and obtain information. Examples of this could be classmates sharing their knowledge in certain subjects, friends recommending each other films to watch or people commenting recent news related to their common hobby.

Related to information sharing there is the concept of *filtering*. In communities there is a large amount of information being shared and not all the information is of the same quality or interest, nevertheless individuals in communities always seem to obtain the information they want. That is the reason why we believe that in communities there exists some kind of information filtering mechanism that makes that high quality and interesting information becomes wide-spread in the community, and on the other hand, low quality information is not spread in the community.

The question we aim to answer with this project is *How do communities share and filter the information?*. We believe that giving an answer to this question would be a big step forward in the field of social networks as it could help in the design of new and better algorithms for social networks, for example a recommendation system that suggests other users that you should connect to in order to get better information. This new algorithms could also help to develop new social networking applications that take advantage of the new discoveries, thus being much more efficient.

1.2 Structure

The rest of the project will be structured as follows. Chapter 2 will consist in the definition and explanation of some hypotheses we believe to be true, all of them backed up by a combination of logical reasoning and real-life examples where they show up. These hypotheses will be some properties/statements of communities in social networks that try to explain how the sharing

and filtering is done. Chapter 3 will present the analysis we have performed using data from real communities in Twitter and the results/conclusions we obtained from this analysis. With this analysis we will try to determine whether the behavior we expected in Chapter 2 is what actually happens in real Twitter communities. Chapter 4 will present an existing mathematical model that closely matches the observations we made in the analysis in Chapter 3. Using this model we derive an interesting result regarding information filtering in communities. Finally, a conclusions chapter where we will do a recap of what we have and haven't achieved with this project and what could be done as future research. Moreover, in the appendices of the project, there will be some extensions to Chapter 3, providing more results and details of the setup used for the experiments.

Chapter 2

Hypotheses

In this chapter, the hypotheses that we have come up with will be presented. Each one of them is an statement that we believe to be true. The purpose of these hypotheses is to help us better understand communities in social networks and try to explain how sharing and filtering is done. Additionally, each hypothesis will be backed up with the reasons that make us think it is true, which will be a combination of logical reasoning and real-life examples where it shows up.

2.1 Preliminary definitions

Before introducing the very first hypothesis of this project we will begin defining a few concepts that will be used throughout the duration of the project.

Definition (Individual): Person or organization that is part of a social network. Where an organization could for example be a company, an sports team or an NGO.

Definition (Community): Group of individuals that interact with one another and share some common characteristics. These common characteristics can be a wide variety of things, for example an ideology or the fact that they live in the same geographical area. In the context of information sharing/filtering that we want to study, this common characteristic will be the interest in a particular topic.

Definition (Member of a community): Individual that is part of a community.

With these definitions, now we are ready to present the first hypothesis of the project.

2.2 Existence of communities

This first hypothesis may seem a bit of a step back, but we believe it is important for a good project to start from the very basics and progress towards its objectives step by step, without jumping to conclusions too fast.

1 Hypothesis *Communities exist in social networks.*

The reasons that make us think that this hypothesis is true will be exposed in the next paragraphs.

As can be observed in society, different individuals may have different interests. Some individual can be interested in basketball, math and music; while other individual can be interested in soccer and cinema. As a consequence of this, different individuals may also want to consume different information, in particular they would like to consume information that is related to their topics of interest.

Without communities, all people with different interests would be mixed up. In that situation, individuals would share information with other individuals that may not have the same interests and, reciprocally, individuals would receive information that is coming from others with potentially different interests.

So communities are a classification of the individuals that are part of an information network in a way that is helpful to them because it helps them to get the information they are interested in.

Moreover, in real life it is easy to observe the existence of communities, they can be present in the form of for example forums, university social clubs or research communities.

Now that we are convinced that communities exist, the next step is to come up with hypotheses that explain more about how information is shared/filtered in communities.

2.3 Information in communities

During our observation of communities we noted that not all of the information being shared is of the same type, we could observe some clear differences in different pieces of information. In particular, information could be classified in the following categories:

- Facts: Pieces of information that explain objectively real-life events. An example of this type of information could be the result of an NBA match.

- Opinions: Pieces of information that express the beliefs of individuals. An example of this type of information could be a user explaining who has been his favourite player in an NBA match.

- Fake information: Pieces of information that resemble facts. However, there is evidence showing that the statements exposed are not true. These type of information might be originated with a malicious purpose to fool individuals or may be originated by the ignorance of individuals themselves. An example of this type of information could be a user posting incorrectly the result of an NBA match.

- Rumours: Pieces of information that expose real-life events that have still not been confirmed nor denied by reliable sources, being reliable sources the only ones that are able to determine whether the exposed event is true or not. After those reliable sources have confirmed or denied a rumour, the rumour becomes either a fact or fake information. An example of this type of information could be some news media announcing the new coach of an NBA team when there has been no official announcement from the team yet.

2.4 Optimality of communities

The next hypothesis defines the optimality criteria that we believe communities satisfy, giving the reasoning that made us come up with it.

2 Hypothesis *Communities are optimal in the sense that:*

1. *Communities provide its members with the most important facts and opinions relevant to the interest of the community.*
2. *Communities minimize the amount of fake information and false rumours received by its members.*
3. *Communities provide its members with the information in (1) as fast as possible.*
4. *Communities provide its members with the information in (1) without redundancy (i.e. only once).*

As a member of a community, all the properties described in the optimal criteria are desirable to have.

First of all, it is logical that members of a community want to be aware of what is happening regarding the topic of interest of the community they participate in, so they want to get the most important facts and opinions relevant to the interest of the community.

Second, obviously fake information is not desirable to members of a community because they want to be well informed, specially lately seeing that the spreading of fake information in social networks is believed to have caused some impact in the result of US elections and Brexit vote [5] [3]. By the same reasoning, members of a community do not want false rumours either, because at the end of the day they are fake information too.

Third, members of a community want to receive information as soon as possible. Who would like to be the last one to be aware of something? If information takes too long to arrive it could

even be the case that the information is no longer valid or it is not useful anymore e.g. finding out about an event once it has taken place.

Finally, there is no point in receiving information more than once, thus members of a community do not want this to happen.

Then, if each of the members of a community wants to have these properties, they will make the best they can to achieve them, making the community optimal as a consequence.

2.5 Content filtering in communities

One key feature of communities that makes them optimal is the one that we will present in the next hypothesis. It is usually taken for granted, however it is not trivial and that is why we think it is best to state it as an hypothesis.

3 Hypothesis *Communities have a filtering mechanism.*

By *filtering mechanism* we mean that communities have some way of discriminating the important information from the not-so-important information in a way that members of the community will not require a lot of effort to find what they are interested in. The filtering mechanism can work in a wide variety of ways, for example by not letting non-important information enter inside the community, by only sharing the important information or by flagging important content in some way that is easy to recognize, thus making members ignore non-important content. The filtering mechanism can also be a combination of the previous or a completely different system.

This hypothesis comes almost as a direct consequence of the previous one: without a filtering mechanism communities would not be optimal, in particular they would not satisfy property (2) of the optimal criteria.

We can observe filtering mechanisms in real-life examples. In the research community, when a researcher wishes to publish a paper with the progress he has made in his research area he

sends this paper to a scientific journal where a decision is made whether the paper is published or not in the journal. This way, bad quality content is rejected and it is not received by other researchers.

In a forum, even if it is not as clear as in the research community, there is also a filtering mechanism. The *reputation* of a member in the forum can work as a way for others to differentiate the quality of the content: the higher the reputation of the member the higher probability that the information that he is posting is of high quality. Note that this reputation measure does not have to be explicit, each member can get an estimate of the reputation of other member based on its number of posts, its time since registration, the quality of its previous posts and other factors. In a forum, filtering is also done via feedback from other users. If some member of the forum posts some information that is not true he will probably get replies telling that what he is saying is not true, or if he posts something that is not interesting he will get replies telling that it is not interesting, it could even be the case that he does not get any response, which will be interpreted as the content that he posted was not interesting. On the other hand, if the same member posts something interesting or useful he will get positive feedback which is a way of flagging the content as good content. So we can say that filtering in a forum works by two different mechanisms: reputation and feedback.

2.6 An optimal community

In this section we are going to describe the structure and functioning of an optimal community (in the sense that satisfies the optimal criteria defined in Hypothesis 2) under the *broadcast model*. In this model (which is the one used by Twitter), users can *follow* other users in order to receive the information they share. So each user has two sets associated to it: the *followers set*, which is the set of users that follow him and the *friends set*, which is the set of users he follows. A detailed proof showing how the community we describe satisfies the optimal criteria will also be given.

Consider a community with the following network structure: there is a member of the commu-

nity that does the filtering, this member is followed by all the other members of the community and, the other way round, this member follows all the other members of the community. We will refer to the member that does the filtering as the *central member* and to the rest of the members as *non-central members*. Then, the network structure of the community looks like Figure 2.1, we will call this structure the *star structure* because of its resemblance with one.

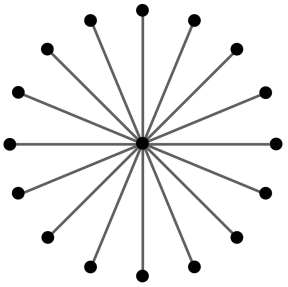

Figure 2.1: Community with star structure

The functioning of this community is as follows. When a non-central member wants to share some information with the rest of the community he posts it so that the central member sees it (because the central member follows everyone) and decides whether it is good enough (according to the optimal criteria) to share it with the rest or not. If the central member considers the information good enough he then forwards it to the rest of the non-central members thus making it available to everyone in the community, otherwise the information gets filtered and is not made available to the rest of the community. Keep in mind that the central member can also share his own content without having to be originated in non-central members.

Observe how this community satisfies all the properties of the optimal criteria.

First, the network structure of the community is a tree so if some information is generated in a member of the community there is only one possible path for that information to get to any other member of the community, then, assuming that members do not share the same information more than once, we get no redundancy.

Second, the information is transmitted in at most two steps, which is optimal because the only way to achieve it in one step would be changing the network structure to a complete graph which is not possible because we would be losing the previous property of no redundancy.

Finally, properties (1) and (2) are consequence of the filtering that is done by the central member.

However, if the community is large enough (lots of non-central members) it is very likely that the central member would have to deal with more information than what he could handle. To solve this problem we can distribute the filtering task among various central members. In this case, each central member would be responsible of following a fraction of the non-central members in a way that every non-central member is at least being followed by one central member and, same as before, non-central members follow all the central members. These central members are connected with each other. This way, when some non-central member has some interesting information one or more central members see this information, but only one of them shares it with the rest of the community, because the other central members will not share it again seeing that other central member has already done it.

Note that making this change from one central member to several of them does not lose optimality because information is still transmitted in two steps and it still satisfies properties (1) and (2) because the central members are still doing the content filtering. Finally, the no redundancy property keeps holding if each of the central members is aware of what the rest of the central members are sharing, because of the reasoning we explained in the previous paragraph.

Observe that this construction achieves the optimal properties for non-central members at the expense of the central members, who will get a lot of bad quality information.

One important question remains to be answered: What are the special properties these central members have that puts them in that position? The next hypothesis gives a reasonable answer to the question.

4 Hypothesis *The central members of an optimal community are members that have high knowledge, experience and reputation in the community topic. They are also very active.*

The central members of an optimal community are the ones that do the filtering, so they require some special abilities. In particular they have to have enough knowledge and experience in the topic to be able to filter fake information and false rumours and also to determine which are

the most important facts and opinions.

They also need to have high reputation so the members of the community trust them to be the ones that provide the information for the community. Without having enough reputation, the members of the community might switch to another central members that they trust more, even if that centrals members have lower knowledge on the community topic.

Finally, they also have to be very active, otherwise, non-central members might not want them to be the central members as they might not be providing the community with enough information.

An interesting aspect of these optimal community we proposed is the resemblance with the properties that Arnau described, where there was a core that did the filtering [7]. In this case, Arnau's core would be the central members.

Chapter 3

Experimental Results

In this chapter, we will present the analysis we have performed on data from real Twitter communities. The first section of the chapter will explain the methodology we have used in order to obtain the users from the communities. The second section will explain the experiments we have performed and how we have performed them. Finally, in the last two sections we will explain the conclusions we can extract from this analysis and a possible explanation for all the observations that we have made.

3.1 Community Sampling

The first approach we used to find members of a community was by manual inspection. In particular, suppose we want to find members of, for example, the fishing community. To do so, we start with a user u_0 that we know belongs to the community. We can find this user using our own knowledge of the community or, in case that we are not able to identify some user for ourselves, we can use the Twitter search tool feeding it with some keyword related to the community, in this case we could use the keyword *fishing*. Once we have this initial user, the following step is to get a set of n users that we will consider to be a sample of the community. We do this using the following algorithm: starting from u_0 we look for another user u_1 that is being followed by u_0 and that seems to be in the community too, to we make sure of this we

inspect u_1's timeline and see if it has tweets related to the community, having u_1 we now look for another user u_2 that is a follower of u_1 and that seems to be in the community too. And so on until we have reached the n users we aim for. Important to note the alternation between followed and follower, we do this to make sure that we get a variety of users in the community and not just "top users" or "bottom users". We call this algorithm the *up-down process* and, similarly, we say that the final n users form a *following-follower chain*, like the one we see in Figure 3.1.

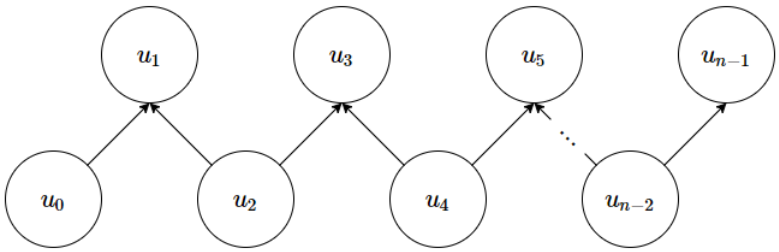

Figure 3.1: Following-follower chain

The choice of n is an important decision to make. On the one hand, the bigger the n the better because we will have a bigger i.e. better sample of the community; on the other hand, we have to keep in mind that this is a manual process, so a huge n is unfeasible because it would take too much time. In our case, we decided to use $n = 60$ because we believe it offers a good compromise: neither too small, nor too large.

Using this algorithm we felt like we had a good random sample of the community, however we also felt like we were missing some important users of the community. To fill this gap, we extended the algorithm we defined previously with an iterative algorithm. This iterative algorithm starts with S_0, the set of n users that we obtained at the end of the up-down process and the iterations go as follows: given S_k, the new set S_{k+1} will be formed by the n users that are most followed by users in S_k. The algorithm stops once it finds a set S_m that is equal to some set S_{i-1} for $i \leq m$, which means that if the algorithm kept running it would have entered in an infinite loop. Finally, we consider our community sample of the community to be $C = \cup_{j=0}^{m} S_j$.

A priori, it is not immediate to see that this algorithm is really giving us users from the same

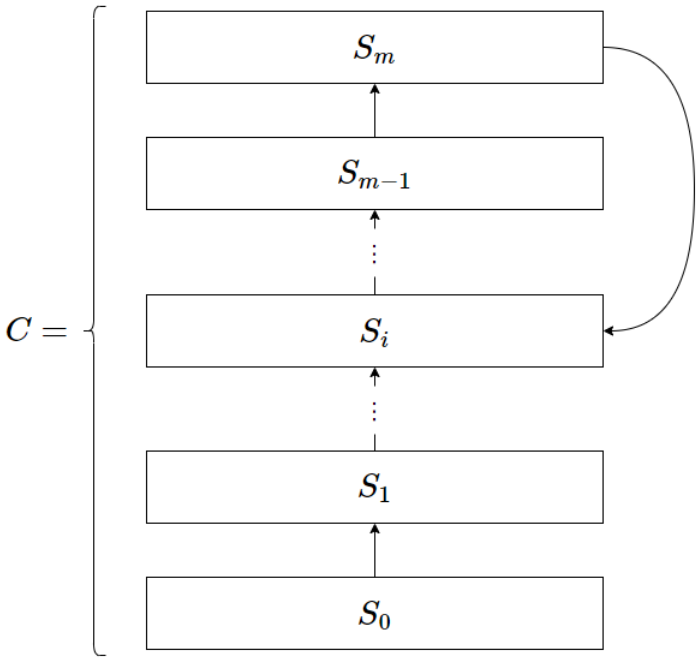

Figure 3.2: Illustration of the iterative algorithm

community: it is possible that users in S_{k+1} are just people that are very followed in Twitter, thus very followed in S_k, but not really from the same community that S_k users are. We refer to this users as *celebrities*. A solution we first explored to prevent this issue was to establish a *celebrity threshold* so that we would not add users to S_{k+1} with more total followers than the celebrity threshold as we considered that they might be a celebrity. We later discovered that this was not needed because, in fact, given that n was somewhat big, if S_k were users from a community then S_{k+1} were also from the community. This is another reason why choosing a bigger n is important.

Another important feature of the algorithm is that it always converged quite quickly: in our tests it converged in at most $m = 24$ iterations. This, added to the fact that the final set C is always pretty small compared to the maximum size it could potentially have (C can potentially be of size $m \cdot n$, however we found communities of sizes around $140 \ll m \cdot n = 24 \cdot 60 = 1440$), which implies that most of the users are repeated along the iterations suggesting that we are finding users that form a community, makes us think that this is a good algorithm to get a sample of users in a community.

3.2 Results

Using the algorithm to find members of a community that we described in the previous section, we sampled the following communities: Artificial Intelligence (AI) community, Haskell community and photography community. In this section, we will present the results and observations using the AI and Haskell communities, the results for the photography community (that look very similar to the ones we will see in this section) can be found in Appendix B.

The first observation we made is that the final set of users we obtained when running the algorithm, i.e. S_m, is highly connected. Figure 3.3 shows an histogram of the number of followers among this 60 users, note that here we are only counting followers inside the group of 60.

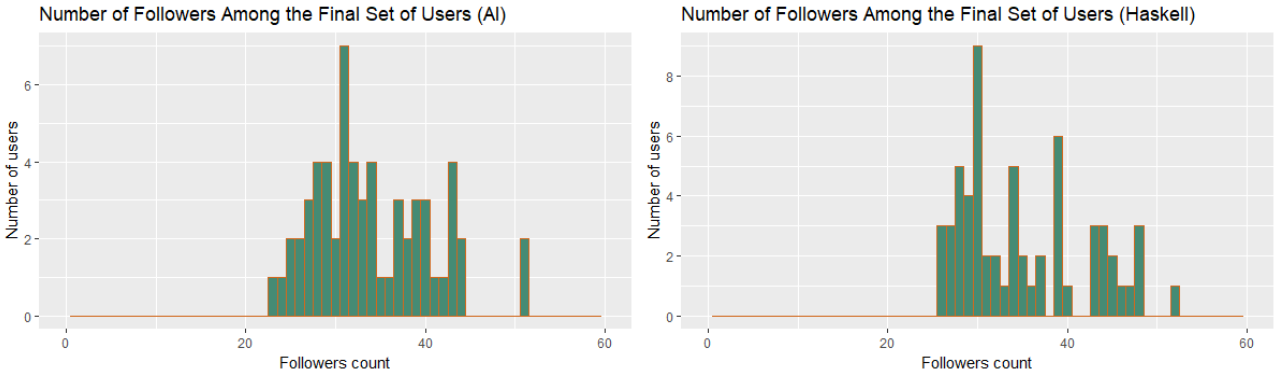

Figure 3.3

As we can see, the number of followers roughly follow a Gaussian distribution with expected value of around 30 i.e. half of the users. Moreover, no user is followed by less than 20 users i.e. a third of the users.

Additionally, to get a better picture of how these groups of users were connected we represented these 60 users for both communities and obtained Figure 3.4. Even though this visualizations don't prove anything, they give us an idea of how strongly connected these sets of users are.

Taking into account Figure 3.3 and Figure 3.4, we believe it is safe to say that S_m is highly connected.

The second observation is that in communities there always appears a small subset of users that

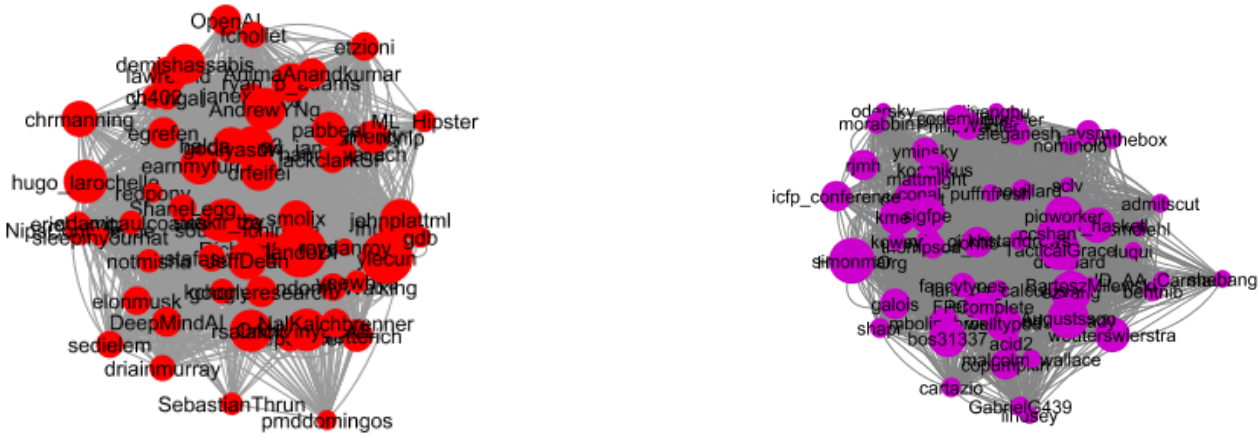

Figure 3.4: Visualization of S_m for AI (red) and Haskell (purple) communities.

are followed by a large amount of users in the community. In particular, there is always a user that is being followed by at least 75% of the users from the community. Figure 3.5 shows for the AI and Haskell communities the histogram of the number of followers among the users in the community. As we can see, in both cases there are some users that are followed by around 100 out of 125 users and 120 out of 150 users, respectively.

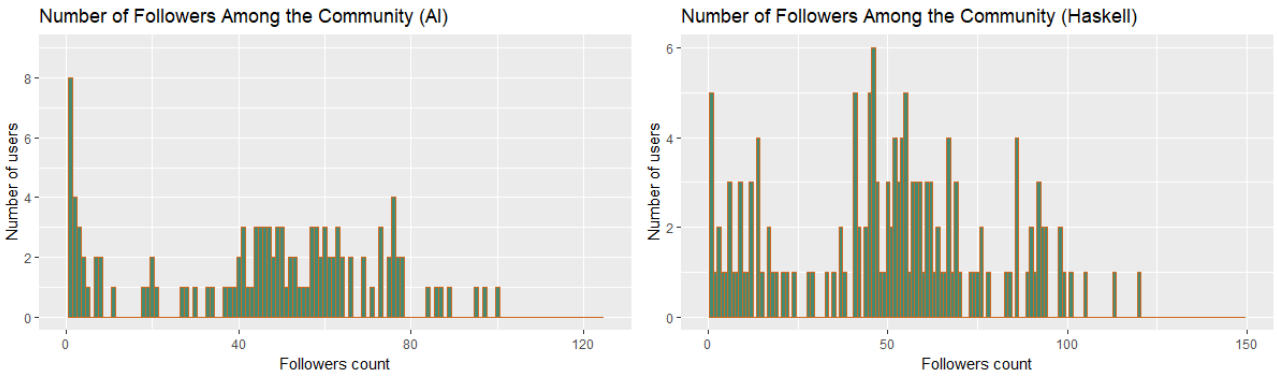

Figure 3.5

For the next observations/results we will be ranking the users in a community. The idea is to rank the users in a way so that the more beneficial a user is for the community, the higher the rank we will give him i.e. high ranked users will be the ones that frequently share good information. The approach we take to rank the users is by number of followers inside the community: the user that has the most number of followers will be assigned rank 1, the user that has the second most number of followers will be assigned rank 2, and so on. Despite being a very simple method to rank the users, this method has proven to give good results.

Regarding the content (tweets) that is distributed through a community there are a few questions that we would like to answer. In particular, we are interested in answering the questions: *Which users produce popular content?*, *Is popular content exclusively produced by high ranked users?* and *Do high ranked users play a key role in making content popular?* (by *popular* we mean that it is widely spread in the community).

To give an answer to these questions we did a little experiment involving the tweets that users in a community posted. We took all the tweets posted by all the users in the community and separated them in two groups: tweets with a high number of retweets inside the community and tweets with a low number of retweets inside the community. This two groups represent popular tweets and non-popular tweets, respectively. Having that, for each group we computed the histogram of the rank of the users that posted the tweets. The results are shown in Figure 3.6.

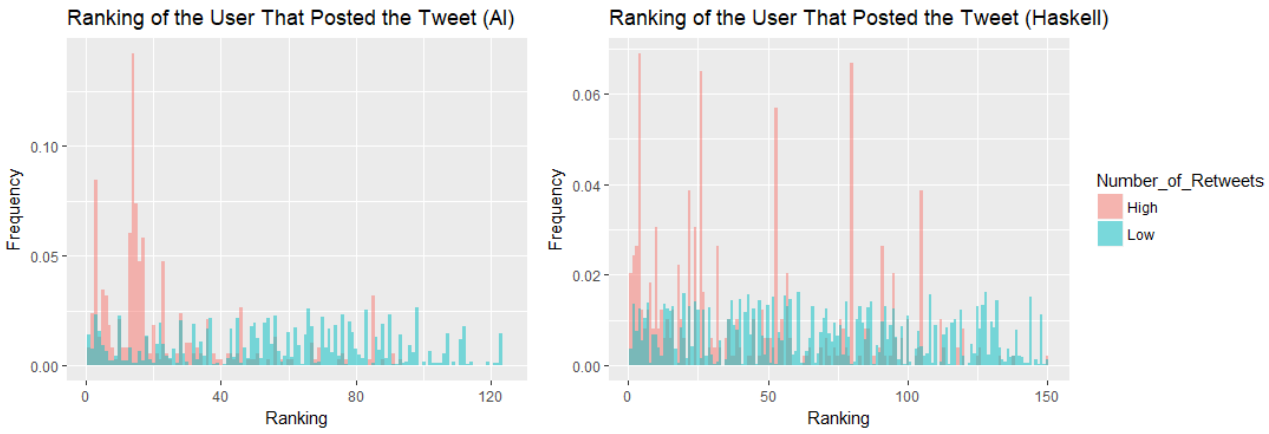

Figure 3.6

As we can see, in both communities, the production of non-popular content (i.e. tweets with low number of retweets) seems to be distributed all across the users of the community (of course, there are some differences in the quantity of production, even some users that do not produce at all, but most of the users are contributing); on the other hand, the production of popular content is much more concentrated in just a few users. Moreover, in the case of the AI community these users that produce most of the popular content are high ranked, which makes sense because as we said high ranked users are the ones that provide the community with frequent good quality content (and higher quality content should be popular in the community).

To complement the information that Figure 3.6 provides us, we did a similar experiment but this time instead of looking at the rank of the user that posted the tweet we are looking at the rank of the highest ranked user that retweeted that particular tweet. Results are shown in Figure 3.7.

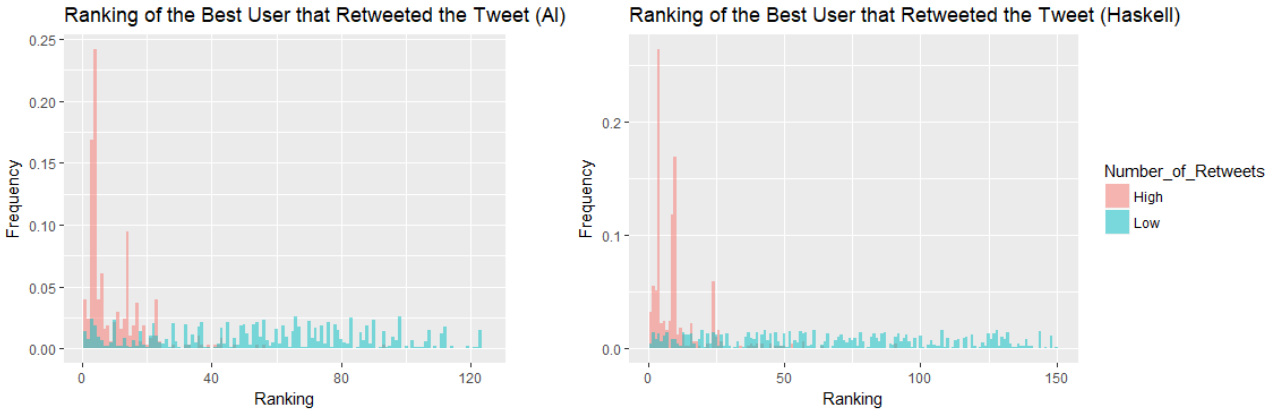

Figure 3.7

The graphics looks very similar to the ones we had before, however, the concentration of popular content is now only in the high ranked users. This means that *all* of the popular content is at some point retweeted by a high ranked user. So, combining the information we have from both Figure 3.6 and Figure 3.7 we can summarize it by saying that: non-popular content is produced evenly across the users in the community; on the other hand, popular content is produced by a smaller group of users, these users are not necessarily high ranked users, however this content only becomes popular when a high ranked user retweets it.

With the information that these two graphics provide us we can answer the questions we formulated before. Popular content is not produced by everybody in the community: only a small group of users can produce popular content. This small group of users has high ranked users as well as lower ranked users. In terms of the role of high ranked users in making content popular we have found that content only becomes popular if a high ranked user has retweeted it at some point, however not all the content they retweet at some point is popular.

Finally, we observed that lower ranked users tend to follow less users and higher ranked users tend to follow more users. Figure 3.8 shows for AI and Haskell communities an histogram of the number of friends (users they follow) for two groups of users: low-ranked users and high-ranked

users.

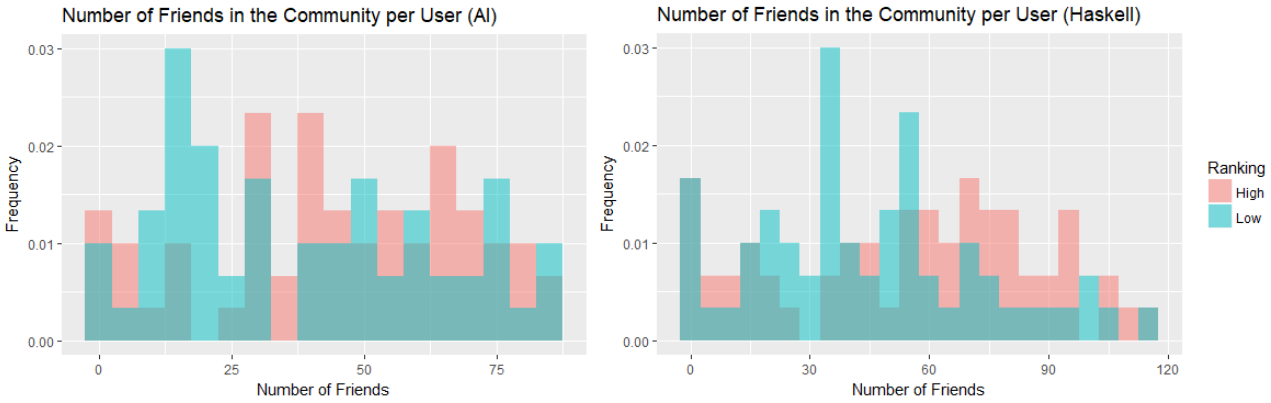

Figure 3.8

As we can see, for low-ranked users the concentration of users is more towards low number of friends, whereas for high-ranked users the concentration is more towards high number of friends. Even if it doesn't seem very clear in these two cases we still believe this to be true. The result for photography community, that can be found in Figure B.6 on Appendix B supports this observation much more clearly.

3.3 Conclusions

Given the observations that we have made by analyzing the data that Twitter communities provided, we believe we can extract the following properties of communities:

1. **There exists a highly connected subset of users.**

2. **There exists a small subset of users that is followed by most of the users in the community.**

3. **Content is produced all across the users in the community, however, the ability to produce popular content is condensed in a few users. Moreover, this content becomes popular only if it is shared by a high ranked user.**

4. **Low ranked users tend to have a lower number of connections in the community, on the other hand high ranked users tend to have a higher number of connections in the community.**

3.4 Possible Explanation

Knowing now these properties of communities, in this section we provide a possible structure for communities that would explain all the properties we observed. We believe that in communities exist the following subsets of users:

1. **ID:** Small set of users that are very recognizable even for users that are not very involved or do not have much knowledge of the community topic.
2. **Core:** Set of users (containing the ID) that have a lot of knowledge and are very involved in the community topic. This group of users provide most of the information in the community.
3. **Outsiders:** Set of users that are not very involved or do not have much knowledge of the community but still want to know the most important information of the community.

Having these groups, the interaction among them would be as follows: Core users (including ID) would follow each other as they are very involved in the community and want to be aware of all the sources of information from the community. Outsiders do not have much knowledge of the community, so they would only follow the most recognizable users i.e. ID users.

Observe how this model would explain each of the properties we observed communities had. In particular, (1) would be explained by the fact that core users connect with each other, thus forming a highly connected set of users. The fact that ID users are followed by core users and outsiders would explain the appearance of users that are followed by almost every user in the community. Property (3) would be explained by the fact that core users all produce content, however it only becomes popular/visible to the majority when ID users retweet it. Finally,

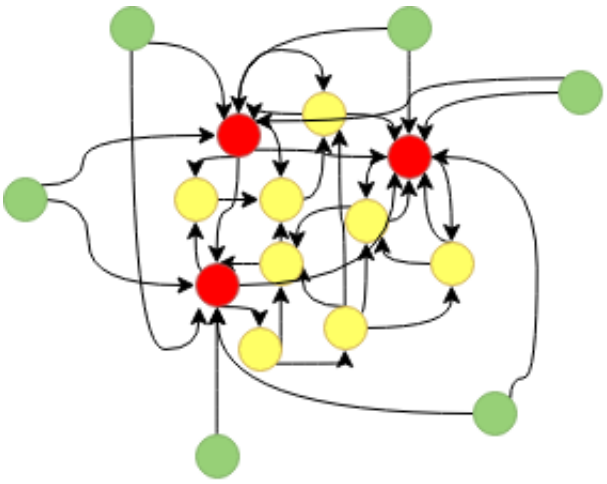

Figure 3.9: Example of network following this model. ID users in red, regular core users in yellow and outsiders in green.

property (4) would be explained by the fact that outsiders (i.e. low ranked users) only follow ID users, which is a small group compared to the core (i.e. high ranked user) that follow among themselves.

Chapter 4

Mathematical Model

In this chapter, we are going to present an existing mathematical model for studying the spread/adoption of trends in social networks [8]. The term *trend* here is used in a broad sense, where a trend could for example be a fashion trend, a new technology, an ideology or a behavior.

In the first two sections of the chapter, the model and the main result that can be proven under this model will be presented. The third section offers a proof of the main result and some other properties. Finally, in the last section we will show the similarities between this model and the observations that we have made of real Twitter communities in the previous chapter and how can we apply the result provided by this model to the problem of information sharing/filtering that we are studying.

4.1 The model

Before starting to explain the model we will define the concept of *active* and *non-active* individuals. In the context of trend adoption, we will refer to an individual who has adopted the trend as active and to an individual who has not adopted the trend as non-active. Individuals can only go from non-active to active status.

4.1.1 The Social Network Graph

The model considers two main groups of individuals within the social network. The first group is the *informed adopters* group, consisting of individuals who have knowledge about the trend and their decision is based on information/discussion with other informed adopters who already follow the trend. This group can also be referred to as the *insiders*. The second group is the *followers* group, consisting of individuals who don't have enough knowledge about the trend to make informed decisions and, as a consequence, they decide whether or not to adopt the trend by imitating their informed adopter acquaintances.

To represent the social network, the model uses a graph G. Inside this graph G, there are considered two subgraphs: an undirected subgraph G_1 and a directed subgraph G_2, both following an Erdös-Rényi random graph model in the way we will precise later. These subgraphs represent the insiders and the followers respectively.

More precisely, the vertex set of G_1 is given by $V(G_1) = \{1, 2, 3, \ldots, n_1\}$, being n_1 the number of informed adopters in G.

Between any pair of vertices $i, j \in V(G_1)$ there exists an undirected edge e_{ij} $(= e_{ji})$ with probability $p_1 \in (0, 1]$ independent of everything else. Then, the (random) edge set of G_1 is given by $E(G_1) = \{e_{ij}\}$.

The average vertex degree in G_1 (i.e. the average number of informed adopters another informed adopter communicates with) is then given by $\lambda_1 = p_1(n_1 - 1)$, which leads to the relation:

$$p_1 = \frac{\lambda_1}{n_1 - 1}. \tag{4.1}$$

Now for the followers group, the vertex set of G_2 is given by $V(G_2) = \{1, 2, 3, \ldots, n_2\}$, being n_2 the number of followers in G.

Between any pair of vertices $i \in V(G_2)$ and $j \in V(G_1)$ there exists a directed edge e_{ij} with probability $p_2 \in (0, 1]$ independent of everything else. Then, the (random) edge set of G_2 is given by $E(G_2) = \{e_{ij}\}$.

The average vertex degree in G_2 (i.e. the average number of informed adopters a follower communicates with) is then given by $\lambda_2 = p_2 n_1$, which leads to the relation:

$$p_2 = \frac{\lambda_2}{n_1}. \tag{4.2}$$

Observe that these two graphs G_1 and G_2 (hence G) are characterized by the parameters (n_1, λ_1) and (n_2, λ_2), given that we have (4.1) and (4.2). To emphasize this dependence of G_1 and G_2 on its parameters, the notation used will be $G_1(n_1, \lambda_1)$ and $G_2(n_2, \lambda_2)$.

Moreover, let $e(G) = |E(G)|$ be the random variable counting the number of edges in G and let $N(i) = \{j \in V(G) | e_{ij} \in E(G)\}$ be the set of neighbors of i.

4.1.2 Asymptotic Behavior

As we are interested in the asymptotic behavior as the social graph becomes large, the model considers an infinite sequence of social graphs $G_1(n_1, \lambda_1)$ and $G_2(n_2, \lambda_2)$ indexed by $n = 1, 2, \ldots$, letting the parameters (n_1, λ_1) and (n_2, λ_2) be functions of n. In the following, we will make some assumptions regarding this parameters.

Assumption 1. *$n_1(n)$ and $n_2(n)$ satisfy:*

$$\lim_{n\to\infty} n_1(n) = \infty \tag{4.3}$$

$$\lim_{n\to\infty} n_2(n) = \infty \tag{4.4}$$

and

$$\lim_{n\to\infty} \frac{n_2(n)}{n_1(n)} = 1. \tag{4.5}$$

Assumption 1 basically means that n_1 and n_2 become as large as we want them to be and they do it at the same rate.

Assumption 2. *We have:*

$$\lim_{n\to\infty} p_2(n) = \lim_{n\to\infty} \frac{\lambda_2(n)}{n_1(n)} = 0. \tag{4.6}$$

This second assumption states that followers can only observe a vanishingly small fraction of the insiders. Note how this is a realistic assumption: even if Earth's population keeps growing, that doesn't mean that the number of acquaintances a particular individual has keeps growing, in fact the number of acquaintances remains almost constant.

4.1.3 The Trendsetters

A third group of individuals that considers the model and we didn't mention before is the *trendsetters*. The trendsetters are the first individuals that adopt the trend. They do it without being influenced by others and serve as the seeds that enable the trend to start spreading.

The model considers the trendsetters to be a subset $A_1 \subseteq V(G_1)$.

Assumption 3. *There exist positive constants ϵ, k and $c \in [0, \frac{1}{2})$ such that:*

$$\lim_{n\to\infty} |A_1(n)| > \epsilon \tag{4.7}$$

and

$$\lim_{n\to\infty} |A_1(n)| \leq k \cdot n_1^c. \tag{4.8}$$

This Assumption 3 represents a restriction in the size of A_1. In particular, the reasoning behind restriction (4.8) is that we want to study the case where A_1 does not dominate G_1.

4.1.4 Trend Adoption Process

The trend adoption process proceeds as a discrete time. At $t = 0$, only the trendsetters have adopted the trend, then the trend starts spreading from A_1 as follows. When a vertex $i \in V(G_1)$ first becomes active at time t, it has a single chance to activate each of its currently non-active neighbors $j \in N(i)$, and it succeeds activating j with a probability ρ_1 independent of everything else. If it succeeds, then vertex j will become active at time $t + 1$ and we will say that edge e_{ij} is an *open edge*. This process, which we will refer to as the cascade process, terminates when there are no new activations from one time step to the next one. On the other hand, followers in G_2 observe their connections in G_1 and adopt the trend as soon as at least t_a friends become active.

4.2 Main Result

The main result that can be proven under this trend adoption model is that the optimal threshold value for the follower group is $t_a = 2$. This threshold value is optimal in the sense that followers never make a mistake by adopting the trend when only a small fraction of the insiders adopt the trend (which can be interpreted as the trend is not worth it, because most of the people that have information/knowledge on the trend haven't adopted it), while maximizing the probability of adopting when a majority of the informed adopters do so (which can be interpreted as the trend is worth it, because most of the people that have information/knowledge on the trend have adopted it).

It can be shown that (asymptotically) a majority of the informed adopters in G_1 will adopt the trend if:

$$\lim_{n\to\infty} \rho_1 \lambda_1 > 1 \tag{4.9}$$

And only a small fraction of informed adopters in G_1 will adopt the trend if:

$$\lim_{n\to\infty} \rho_1 \lambda_1 < 1. \tag{4.10}$$

With (4.9) and (4.10), the result we presented before can be reworded as follows. The threshold value $t_a = 2$ is optimal in the sense that followers never adopt when $\rho_1\lambda_1 < 1$ and maximize the probability of adopting when $\rho_1\lambda_1 > 1$.

Given a follower $i \in V(G_2)$, let K_i be the random variable counting the number of active neighbors of i after the cascade process has finished. Let $F(k) = Pr(\exists i \in V(G_2) : K_i \geq k)$ under the assumption that $\rho_1\lambda_1 < 1$ i.e. $F(k)$ is the probability that there exists at least one vertex in G_2 with at least k active neighbors in G_1, all that under the assumption that we are in the case $\rho_1\lambda_1 < 1$. Then, the proof of the result will consist in showing that:

$$\lim_{n\to\infty} F(1) > 0 \tag{4.11}$$

and

$$\lim_{n\to\infty} F(k) = 0 \qquad \forall k \geq 2. \tag{4.12}$$

On the one hand, proving (4.12) would show that when $\rho_1\lambda_1 < 1$ (i.e. only a small fraction of the insiders adopt) one can't find followers having two or more active insider friends, which means that choosing threshold values $t_a \geq 2$ ensures that followers never adopt when $\rho_1\lambda_1 < 1$ (i.e. followers never make a mistake). On the other hand, (4.11) tells us that when $\rho_1\lambda_1 < 1$ it is still possible to find followers with at least one active insider friend, which means that $t_a = 1$ may lead to followers adopting when only a small fraction of informed adopters do so. In conclusion, using a threshold $t_a \geq 2$ makes sure followers make no mistake by adopting when only a small fraction of the informed adopters do so, moreover, the value $t_a = 2$ is optimal because it maximizes the probability of adopting when a majority of the informed adopters do it.

4.3 Equivalent Model and Proof

The first step to prove the result is to introduce a new model, equivalent to the one that is defined in the previous section. This new model, instead of first considering the social

connections in G_1 with a probability p_1 and then spreading the trend with a probability ρ_1 of succeeding, it considers social connections with a probability $p_1' = p_1\rho_1$ and then the spreading of the trend is over all connections, without a probability of succeeding.

More precisely, the new model G_1' is an Erdös-Rényi graph with the same vertex set as G_1 and independent probability $p_1' = p_1\rho_1$ of having any given edge, being p_1 and ρ_1 the same ones used in G_1, the initial active set is the same as the one in G_1, i.e. A_1, and subject to the same constraints (4.7) and (4.8).

These two models are equivalent in the sense that the distribution in the size of the *influenced sets* (i.e. the set of active individuals in the group of insiders after the cascade process) is the same. In particular, let $I_1 = \{i \in V(G_1) | i \text{ connects to } A_1 \text{ via open edges in } G_1\}$ be the influenced set of G_1 and $I_1' = \{i \in V(G_1') | i \text{ connects to } A_1 \text{ via edges in } G_1'\}$ be the influenced set of G_1'. This first theorem will prove that the distribution in the size of the influenced sets is the same.

Theorem 1. *Given an initial active set A_1, the distribution of the influenced sets obtained by cascade process on G_1 starting from A_1, is the same as the distribution of sets reachable from A_1 via edges on G_1'.*

Proof: First we compute the probability that a given vertex $i \in V(G_1)$ is activated in iteration $t+1$ of the cascade process in G_1. We define $A_1^{(t)}$ to be the set of active vertices at the end of iteration t. Then, the probability that i becomes active for the first time at iteration $t+1$ is equal to the probability that it has a neighbor in $A_1^{(t)}$ but not in $A_1^{(t-1)}$ that succeeds in activating i. This probability is $Pr(i \in A_1^{(t+1)} | i \notin A_1^{(t)}) = 1 - (1 - \frac{\lambda_1\rho_1}{n_1 - 1})^{|A_1^{(t)} \setminus A_1^{(t-1)}|}$.

For G_1' we construct it gradually as follows. Starting from the initial set A_1, for each vertex i that has at least one edge stub, we determine whether it connects to A_1. If that is the case, then i is reachable; if not, it remains to be determined the source of that edge, subject to the condition that it does not come from A_1. This gives us a new set $A_1^{(1)}$ of reachable vertices (in 1 step) from A_1. Repeating this process we construct sets $A_1^{(2)}, A_1^{(3)}, \cdots$. Now, the probability that a vertex i is determined to be reachable in step $t+1$ knowing that it was not reachable

in step t is the probability that some of its edges come from $A_1^{(t)}$ but not from $A_1^{(t-1)}$, this probability is $Pr(i \in A_1^{(t+1)} | i \notin A_1^{(t)}) = 1 - (1 - \frac{\lambda_1 \rho_1}{n_1 - 1})^{|A_1^{(t)} \setminus A_1^{(t-1)}|}$.

Then, by induction, we can see that we obtain the same distribution of influenced sets in G_1 and in G_1'. ■

The main benefit of introducing this equivalent model is that G_1' is an Erdös-Rényi graph and the properties of Erdös-Rényi graphs have been thoroughly studied in previous works [6].

Let P_a be the probability that a uniformly selected random vertex in G_1' is active after the cascade process terminates, P_a is given by:

$$P_a = \frac{|I_1'|}{n_1} \tag{4.13}$$

In order to quantify P_a, we use the expectation of $|I_1'|$. Next theorem will help us to compute this expectation.

Theorem 2. *On an Erdös-Rényi random graph $G(n, p = \frac{\lambda}{n-1})$, the expectation of the size of the influenced set with one vertex active initially is $\frac{1-\rho}{1-\rho-\lambda\rho}$, being ρ the probability that a given vertex succeeds in spreading the trend to a given neighbor.*

Proof: Let p_k be the probability for a given vertex $i \in V(G)$ of having degree k, p_k is given by:

$$p_k = \binom{n-1}{k} p^k (1-p)^{n-1-k} \approx \frac{\lambda^k}{k!} e^{-\lambda} \tag{4.14}$$

Let $G_0(x)$ be the generating function of $\{p_k\}$, which is defined by:

$$G_0(x) = \sum_{k \geq 0} p_k x^k \tag{4.15}$$

Select an edge from $E(G)$ and let j be one of its (two) incident vertices. Let $\{q_k\}$ be the degree distribution of vertex j, and let $G_1(x)$ be the generating function of $\{q_k\}$. If we uniformly choose an edge from $E(G)$ each edge has probability $\frac{1}{e(G)}$ of being chosen. Then, the probability that

a uniformly chosen edge is incident to j, conditional on j having degree k is:

$$Pr(\text{chosen edge incident to } j|deg(j) = k) = \frac{k}{e(G)} \tag{4.16}$$

And by law of total probability this gives us:

$$Pr(\text{chosen edge incident to } j) = \sum_{k\geq 0} \frac{k}{e(G)} p_k = \frac{1}{e(G)} \sum_{k\geq 0} k \cdot p_k \tag{4.17}$$

Applying Bayes rule, we obtain:

$$\begin{aligned} q_k &= Pr(deg(j) = k|\text{chosen edge incident to } j) \\ &= \frac{Pr(\text{chosen edge incident to } j|deg(j) = k)Pr(deg(j) = k)}{\sum_{l\geq 0} Pr(\text{chosen edge incident to } j|deg(j) = l)Pr(deg(j) = l)} \\ &= \frac{k \cdot p_k}{\sum_{l\geq 0} l \cdot p_l} = \frac{k \cdot p_k}{\lambda} \end{aligned} \tag{4.18}$$

Now, let $\tilde{p}_m$ be the probability for a given vertex $i \in V(G)$ of having m open edges, and let

$G_0(x;\rho)$ be the generating function of $\{\tilde{p}_m\}$. We have that:

$$\begin{aligned}
\tilde{p}_m &= Pr(m \text{ edges of } i \text{ are open}) \\
&= \sum_{k\geq 0} Pr(m \text{ edges of } i \text{ are open}|deg(i)=k)Pr(deg(i)=k) \\
&= \sum_{k\geq m} \binom{k}{m} \rho^m (1-\rho)^{k-m} p_k \\
&\Longrightarrow \\
G_0(x;\rho) &= \sum_{m\geq 0} \tilde{p}_m x^m \\
&= \sum_{m\geq 0}\sum_{k\geq m} \binom{k}{m} \rho^m (1-\rho)^{k-m} p_k x^m \\
&= \sum_{k\geq 0}\sum_{m=0}^{k} \binom{k}{m} \rho^m (1-\rho)^{k-m} p_k x^m \\
&= \sum_{k\geq 0} p_k \sum_{m=0}^{k} \binom{k}{m} \rho^m (1-\rho)^{k-m} x^m \\
&= \sum_{k\geq 0} p_k (1-\rho+\rho x)^k = G_0(1-\rho+\rho x) \\
&\Longrightarrow \\
G_0'(x;\rho) &= \rho G_0'(1-\rho+\rho x)
\end{aligned} \tag{4.19}$$

Now, select an edge from $E(G)$ and let j be one of its (two) incident vertices. Let $\tilde{q}_m$ be the probability for vertex j of having m open edges, and let $G_1(x;\rho)$ be the generating function of

$\{\tilde{q}_m\}$. Similar to the derivation we did before, we have that:

$$\begin{aligned}
\tilde{q}_m &= Pr(m \text{ edges of } j \text{ are open}) \\
&= \sum_{k\geq 0} Pr(m \text{ edges of } j \text{ are open}|deg(j)=k)Pr(deg(j)=k) \\
&= \sum_{k\geq m} \binom{k}{m}\rho^m(1-\rho)^{k-m}q_k \\
&\Longrightarrow \\
G_1(x;\rho) &= \sum_{m\geq 0}\tilde{q}_m x^m \\
&= \sum_{m\geq 0}\sum_{k\geq m}\binom{k}{m}\rho^m(1-\rho)^{k-m}q_k x^m \\
&= \sum_{k\geq 0}\sum_{m=0}^{k}\binom{k}{m}\rho^m(1-\rho)^{k-m}q_k x^m \\
&= \sum_{k\geq 0} q_k \sum_{m=0}^{k}\binom{k}{m}\rho^m(1-\rho)^{k-m}x^m \\
&= \sum_{k\geq 0} q_k(1-\rho+\rho x)^k = G_1(1-\rho+\rho x) \\
&\Longrightarrow \\
G_1'(x;\rho) &= \rho G_1'(1-\rho+\rho x)
\end{aligned} \tag{4.20}$$

Let $\mathbf{Z}$ be a random variable denoting the size of the influenced set starting from a given vertex i, and let $H_0(x;\rho)$ be the generating function for the distribution of $\mathbf{Z}$. Let $\tilde{Z}$ be a random variable which indicates the size of the influenced starting from a given edge, and let $H_1(x;\rho)$ be the generating function for the distribution of $\tilde{Z}$.

With this definitions, we have that:

$$\mathbf{Z} = 1 + \sum_k \tilde{Z}^{(k)} \tag{4.21}$$

Where the number of $\tilde{Z}^{(k)}$'s is determined by the number of neighbors the vertex i has, which has distribution $\{p_k\}$. Using (4.21) and the properties of probability generating functions we

have that:

$$H_0(x;\rho) = xG_0(H_1(x;\rho);\rho) \tag{4.22}$$

Similarly for $\tilde{Z}$, we have that:

$$\tilde{Z} = 1 + \sum_k \tilde{Z}^{(k)} \tag{4.23}$$

Where the number of $\tilde{Z}^{(k)}$'s is determined by the number of neighbors the vertex attached to the edge has, which in this case the distribution is given by $\{q_k\}$. Using (4.23) and the properties of probability generating functions we have that:

$$H_1(x;\rho) = xG_1(H_1(x;\rho);\rho) \tag{4.24}$$

Again by the properties of probability generating functions and (4.22) we have that the expected value of $\mathbf{Z}$ is given by:

$$E(\mathbf{Z}) = H_0'(1;\rho) = 1 + G_0'(1;\rho)H_1'(1;\rho) \tag{4.25}$$

And differentiating (4.24) we obtain:

$$\begin{aligned} & H_1'(1;\rho) = 1 + G_1'(1;\rho)H_1'(1;\rho) \\ \Longrightarrow & \\ & H_1'(1;\rho) = \frac{1}{1 - G_1'(1;\rho)} \end{aligned} \tag{4.26}$$

Now, combining (4.25), (4.26), (4.19) and (4.20) we get:

$$E(\mathbf{Z}) = 1 + \frac{\rho G_0'(1)}{1 - \rho G_1'(1)} \tag{4.27}$$

Moreover $G_0'(1)$ and $G_1'(1)$ can be computed as follows:

$$G_0'(1) = \sum_{k\geq 1} kp_k x^{k-1}|_{x=1} = \sum_{k\geq 1} kp_k = \sum_{k\geq 0} kp_k = \lambda \tag{4.28}$$

and

$$\begin{aligned} G_1'(1) &= \sum_{k\geq 1} kq_k x^{k-1}|_{x=1} = \sum_{k\geq 1} kq_k = \sum_{k\geq 0} kq_k \\ &= \sum_{k\geq 1} k\frac{kp_k}{\lambda} = \frac{1}{\lambda}\sum_{k\geq 1} k^2 p_k = \frac{1}{\lambda}E(K^2) \\ &= \frac{Var(K)+E(K)^2}{\lambda} = \frac{\lambda+\lambda^2}{\lambda} = 1+\lambda \end{aligned} \tag{4.29}$$

Being K a random variable denoting the degree of a vertex, which we now follows (approximately) a Poisson distribution, hence $E(K) = Var(K) = \lambda$. Finally substituting (4.28) and (4.29) in (4.27) we get:

$$E(\mathbf{Z}) = 1 + \frac{\rho\lambda}{1-\rho(1+\lambda)} = \frac{1-\rho}{1-\rho-\rho\lambda} \tag{4.30}$$

■

By this theorem we have that:

$$E(|I_1'|) = \frac{1-\rho_1}{1-\rho_1-\rho_1\lambda_1} \qquad \text{if } |A_1| = 1 \tag{4.31}$$

So, for $|A_1| \leq k \cdot n_1^c$ we have that:

$$E(|I_1'|) \leq k \cdot n_1^c \frac{1-\rho_1}{1-\rho_1-\rho_1\lambda_1} \leq k' \cdot n_1^c \qquad \text{for some } k' \tag{4.32}$$

Applying Markov inequality and (4.32) we can obtain an upper bound for $|I_1'|$:

$$Pr(|I_1'| \geq n_1^{2c}) \leq \frac{E(|I_1'|)}{n_1^{2c}} \leq \frac{k' \cdot n_1^c}{n_1^{2c}} = \frac{k'}{n_1^c} \xrightarrow[n\to\infty]{} 0 \tag{4.33}$$

Since $c \in [0, \frac{1}{2}) \implies 2c = c' \in [0, 1)$ and the distributions of $|I_1|$ and $|I_1'|$ are the same by Theorem 1, (4.33) can be rewritten as:

$$\lim_{n\to\infty} Pr(|I_1'| \geq n_1^{c'}) = 0 \tag{4.34}$$

or equivalently:

$$\lim_{n\to\infty} Pr(|I_1'| < n_1^{c'}) = 1 \tag{4.35}$$

Now, we can bound P_a by substituting (4.35) into (4.13), which gives us:

$$P_a = \frac{|I_1|}{n_1} < \frac{n_1^{c'}}{n_1} = 0 \text{ with probability 1 for } n_1 \text{ large} \tag{4.36}$$

(4.36) has a very intuitive interpretation: if we randomly pick a vertex after the cascade process has terminated, then that vertex is non-active with probability approaching 1 for large networks.

Theorem 3. *On the model $G'(n_1, n_2, \rho_1\lambda_1, A_1, \lambda_2, t_a)$, given a vertex $i \in V(G_2)$, let K_i be defined as Section 4.2. Then, $Pr(K_i = n_a) = \frac{(\lambda_2 P_a)^{n_a}}{n_a! e^{\lambda_2 P_a}}$.*

Proof: Let p_k be the probability for a given vertex $i \in V(G_2)$ of having k neighbors in G_1. This probability is given by a binomial distribution with parameters n_1 and $p_2 = \frac{\lambda_2}{n_1}$, so $p_k = \binom{n_1}{k} p_2^k (1-p_2)^{n_1-k}$. Applying Poisson approximation we obtain $p_k \approx \frac{\lambda_2^k}{k!} e^{-\lambda_2}$.

Now, we want to derive the distribution of the number of active neighbors a given vertex $i \in V(G_2)$ has. To do so, we first derive it conditional to i having k neighbors. Then we have $Pr(K_i = n_a || N(i)| = k) = \binom{k}{n_a} P_a^{n_a} (1-P_a)^{k-n_a}$.

Then, by the law of total probability we have that:

$$\begin{aligned}
Pr(K_i = n_a) &= \sum_k Pr(K_i = n_a || N(i)| = k) Pr(|N(i)| = k) \\
&= \sum_k \binom{k}{n_a} P_a^{n_a} (1-P_a)^{k-n_a} p_k \\
&= \sum_k \frac{k!}{n_a!(k-n_a)!} P_a^{n_a} (1-P_a)^{k-n_a} \frac{\lambda_2^k}{k!} e^{-\lambda_2} \\
&= \frac{P_a^{n_a} e^{-\lambda_2}}{n_a!} \sum_k \frac{(1-P_a)^{k-n_a} \lambda_2^k}{(k-n_a)!} \\
&= \frac{P_a^{n_a} e^{-\lambda_2} \lambda_2^{n_a}}{n_a!} \sum_k \frac{(1-P_a)^{k-n_a} \lambda_2^{k-n_a}}{(k-n_a)!} \\
&= \frac{P_a^{n_a} e^{-\lambda_2} \lambda_2^{n_a} e^{\lambda_2(1-P_a)}}{n_a!} \\
&= \frac{(\lambda_2 P_a)^{n_a} e^{-\lambda_2 P_a}}{n_a!} = \frac{(\lambda_2 P_a)^{n_a}}{n_a! e^{\lambda_2 P_a}}
\end{aligned} \tag{4.37}$$

■

Theorem 4. *We have:*

$$\lim_{n\to\infty} F(1) > 0 \tag{4.38}$$

and

$$\lim_{n\to\infty} F(k) = 0 \qquad \forall k \geq 2. \tag{4.39}$$

Proof:

$$\begin{aligned}
\lim_{n\to\infty} F(1) &= \lim_{n\to\infty} 1 - (1 - Pr(K_i \geq 1))^{n_2} \\
&= \lim_{n\to\infty} 1 - Pr(K_i = 0)^{n_2} \\
&= 1 - \lim_{n\to\infty} Pr(K_i = 0)^{n_2} \\
&= 1 - \lim_{n\to\infty} [\frac{(\lambda_2 P_a)^0}{0! e^{\lambda_2 P_a}}]^{n_2} \\
&= 1 - \lim_{n\to\infty} [\frac{1}{e^{\lambda_2 P_a}}]^{n_2} \\
&= 1 - \lim_{n\to\infty} \frac{1}{e^{\lambda_2 P_a n_2}} > 0
\end{aligned} \tag{4.40}$$

Now for (4.39) we will show that it is true that $\lim_{n\to\infty} F(k) = 0$ for $k = 2$ which will imply that it is true $\forall k \geq 2$, because if there are no followers having 2 or more active neighbors, then there are no followers having 3 or more active neighbors, and so forth.

$$
\begin{aligned}
\lim_{n\to\infty} F(2) &= \lim_{n\to\infty} 1-(1-Pr(K_v\geq 2))^{n_2} \\
&= 1-\lim_{n\to\infty}[Pr(K_i=0)+Pr(K_i=1)]^{n_2} \\
&= 1-\lim_{n\to\infty}[\frac{(\lambda_2 P_a)^0}{0!e^{\lambda_2 P_a}}+\frac{(\lambda_2 P_a)^1}{1!e^{\lambda_2 P_a}}]^{n_2} \\
&= 1-\lim_{n\to\infty}[\frac{1}{e^{\lambda_2 P_a}}+\frac{\lambda_2 P_a}{e^{\lambda_2 P_a}}]^{n_2} \\
&= 1-\lim_{n\to\infty}[\frac{1+\lambda_2 P_a}{e^{\lambda_2 P_a}}]^{n_2} \\
&= 1-\lim_{n\to\infty}[\frac{1+\frac{\lambda_2|I_1|}{n_1}}{e^{\frac{\lambda_2|I_1|}{n_1}}}]^{n_2} \\
&= 1-\lim_{n\to\infty}\frac{(1+\frac{\lambda_2|I_1|}{n_1})^{n_2}}{e^{\lambda_2|I_1|\frac{n_2}{n_1}}} \\
&= 1-\frac{\lim_{n\to\infty}(1+\frac{\lambda_2|I_1|}{n_1})^{n_2}}{\lim_{n\to\infty} e^{\lambda_2|I_1|\frac{n_2}{n_1}}} \\
&= 1-\frac{e^{\lambda_2|I_1|}}{e^{\lambda_2|I_1|}} = 1-1=0
\end{aligned}
\tag{4.41}
$$

Where in the last step we have used that:

$$
\begin{aligned}
\lim_{n\to\infty}(1+\frac{x}{n_1})^{n_2} &= \lim_{n\to\infty}(1+\frac{1}{\frac{n_1}{x}})^{\frac{n_1}{x}\cdot\frac{n_2}{n_1}x} \\
&= \lim_{n\to\infty}[(1+\frac{1}{\frac{n_1}{x}})^{\frac{n_1}{x}}]^{\frac{n_2}{n_1}x} \\
&= [\lim_{n\to\infty}(1+\frac{1}{\frac{n_1}{x}})^{\frac{n_1}{x}}]^{\lim_{n\to\infty}\frac{n_2}{n_1}x} \\
&= e^x
\end{aligned}
\tag{4.42}
$$

Which is true given that we assumed (4.5). ■

4.4 Connection with the Model

This model we have just presented shows several similarities with the observations that we had made by analyzing the data of Twitter communities in the previous chapter.

First, this model considers two groups of users: one group of users is the informed adopters, that have knowledge about the trend (or the topic concerning the trend) and their decision whether or not to follow the trend is based in their knowledge and the interaction with other informed adopters that already follow the trend; the second group is the followers, who don't have much knowledge and as a consequence they imitate what informed adopters do. These two groups match closely the properties of the core and the outsiders respectively, as we defined them in the previous chapter.

Another similarity is how these groups interact with each other. As we said, core users would mainly follow (make connections) among them, as informed adopters do in G_1. On the other hand, outsiders would only follow ID users. This property is not fully reflected in this model as individuals in G_2 randomly follow individuals in G_1 which would correspond to core users, but it is closely related.

Now it only remains to determine what is a trend in the context of information sharing/filtering that we are studying. A very reasonable interpretation of a trend in this context could be the following: adopting a trend means to believe some information to be true.

With this interpretation of a trend, the main result of the model would say that in a social network, an outsider has to trust some information if it is shared by at least two core users. We could say that this strategy of believing some information to be true if it is shared by at least two users acts as content filtering mechanism for outsiders to get reliable information in a social network.

Chapter 5

Conclusion

With this project we tried to study how communities shared and filtered the information. Even though the analysis in Chapter 3 does not seem to support the behavior of communities we expected in Chapter 2 it has still been very useful and has provided us with some interesting observations regarding the structure of communities in social networks. Not only that, at the end of Chapter 3 we give a very simple and reasonable model of how communities could work that is able to explain all the properties we observed in the analysis. This model, apart from making a lot of sense, seems to be very closely related to an existing mathematical model that is used to analyze the spread of trends in social networks. In Chapter 4 we study this mathematical model and adapt it to the setting of sharing/filtering that we are interested. Doing that, we obtain an interesting result that proposes an optimal filtering strategy for the outsiders of the community.

5.1 Future Research

The analysis done in Chapter 3 does give us some intutition on what is happening but it is still very superficial, it would be interesting to do more experiments (possibly with a larger number of communities and more diverse) to further validate the observations and the possible explanation that we give at the end of that same chapter.

Another interesting path of future research could be to do an extension of the mathematical model to take into account the fact that outsiders only follow ID users, instead of core users in general as it is right now.

Appendix A

Experiments Setup

In this appendix we explain with more detail the technical setup we have used to download and store the Twitter data that we later used to perform the experiments. Figure A.1 shows the setup we have used.

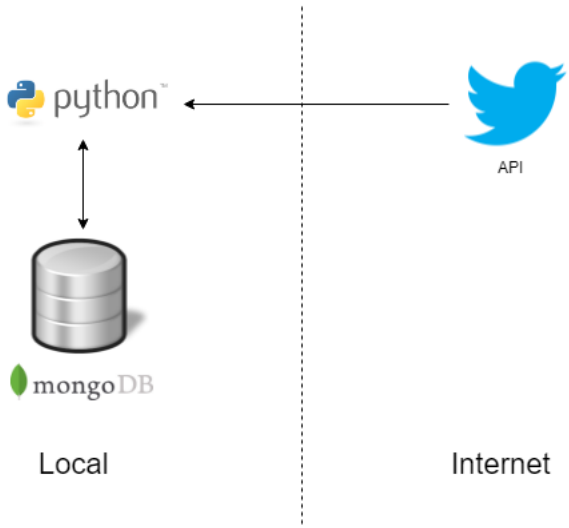

Figure A.1: Diagram of the setup

A.1 Downloading Data

To obtain data from Twitter in an automatized way, Twitter offers an API aimed to developers of applications that wish to create extensions of Twitter [1]. We have used this API to obtain

the data we needed for our experiments. To access the API we used Tweepy, a Python package that acts as a wrapper for the Twitter API [4].

The Twitter API has rate limitations on some of its functions, for example, it is only possible to download 75000 followers every 15 minutes. The main benefit of using Tweepy (apart from the integration with Python) is that it handles this rate limitations and possible network errors in a transparent way to the programmer.

A.2 Storing Data

Given the rate limitations that the Twitter API presents, it is unfeasible to query the API every time we need some information because it would be very slow. The best approach to reduce the amount of queries to the API is by storing the data once we query the API for some information. For example, in the algorithm described in Section 3.1 we need to get multiple times the list of users a given user follows. Using this approach we would only query the API for this list the first time, then the next times we can obtain the list using the stored data that we already have.

To store the data we have used mongoDB [2]. We chose this type of database because of its simplicity and flexibility, allowing us to add/modify fields as we felt we needed them. On top of that, another deciding factor was that we already had some experience using this type of database.

In mongoDB the basic units of storage are JSON documents. These documents are grouped in collections and, at the same time, collections are grouped in databases. The structure we use to store our data is the following: for each community we have a database, each database contains two collections, one for users in which we store all the data from the users in the community and one for tweets in which we store all the tweets that the community users posted or retweeted (limited to 3200 per user by the Twitter API).

As can be seen in Figure A.2 for each user we store: user id, screen name, a boolean saying

Figure A.2: Sample document of the users collection

whether it is a verified user or not, number of followers, number of friends (users they follow), number of tweets, date of registration and a list containing the id of its friends. Two additional fields are later computed and stored: a list containing the id of its friends that are in the community and a list containing the id of its followers that are in the community.

Note that we only store the list of its friends and not the list of its followers. The reason behind this is that very popular users have hundreds of thousands or even millions of followers which would take more than 3 hours per user to download. On the other hand, downloading friends is much faster as most of the users follow at most a thousand of users which takes just a minute to download. Moreover, having only the lists of friends it is enough to compute the list of followers that are in the community, which is something we are very interested in.

In the case of tweets, we store: id of the tweet, text/content of the tweet, id of the user that posted the tweet, the date the tweet was posted, the number of retweets the tweet has and number of likes; additionally, if the tweet is a retweet, we store the id of the tweet that is retweeting. Same as before, two additional fields are computed and stored: the list of the ids

```
{
    "_id" : "920845315779039232",
    "text" : "RT @brendonbrewer: The fundamental theorem of information theory. https://t.co/lohGayZPvK",
    "user_id" : "44073696",
    "created_at" : ISODate("2017-10-19T02:53:26.000Z"),
    "retweet_count" : 145,
    "favorite_count" : 0,
    "retweeted_id" : "920422736656089090",
    "retweets_within_community" : [],
    "retweet_within_community_count" : 0
}
```


Figure A.3: Sample document of the tweets collection

of the users of the community that have retweeted the tweet and the size of this list.

Appendix B

Photography Results

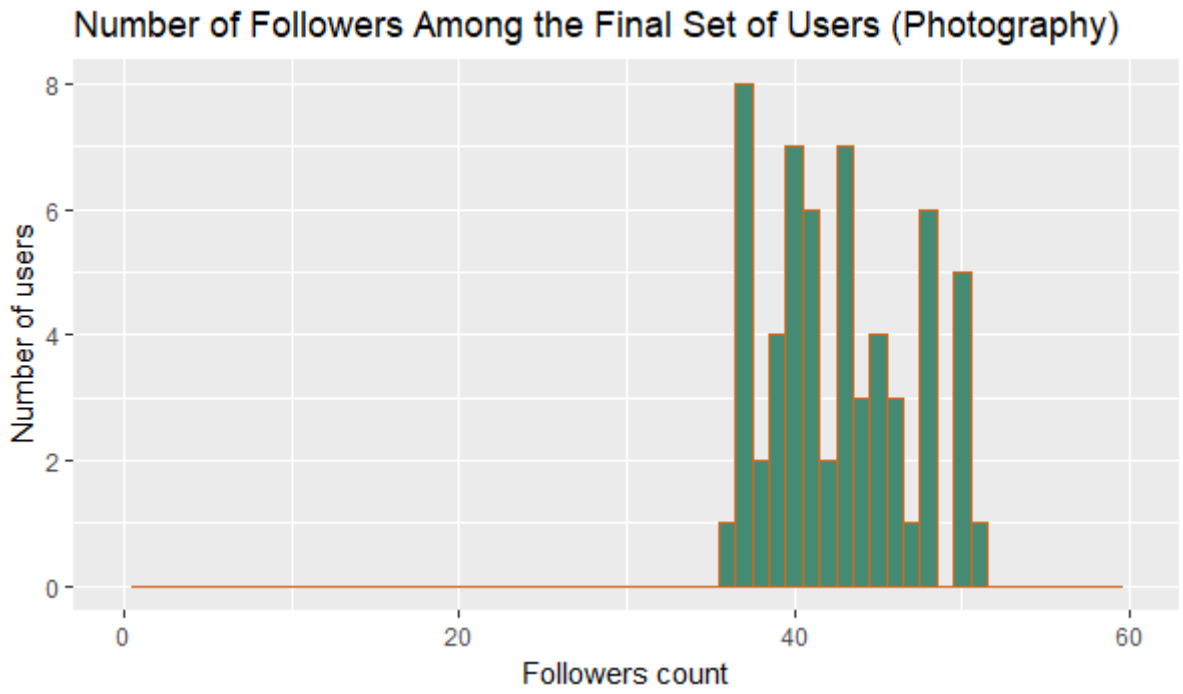

Figure B.1

Figure B.2: Visualization of S_m

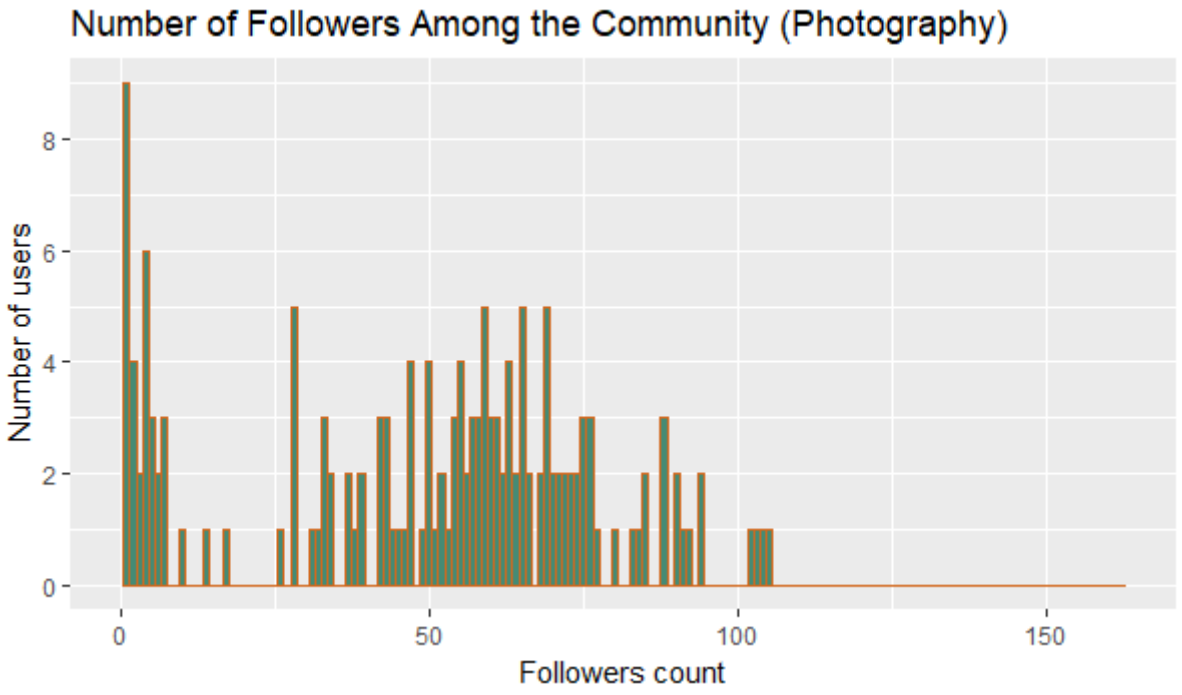

Figure B.3

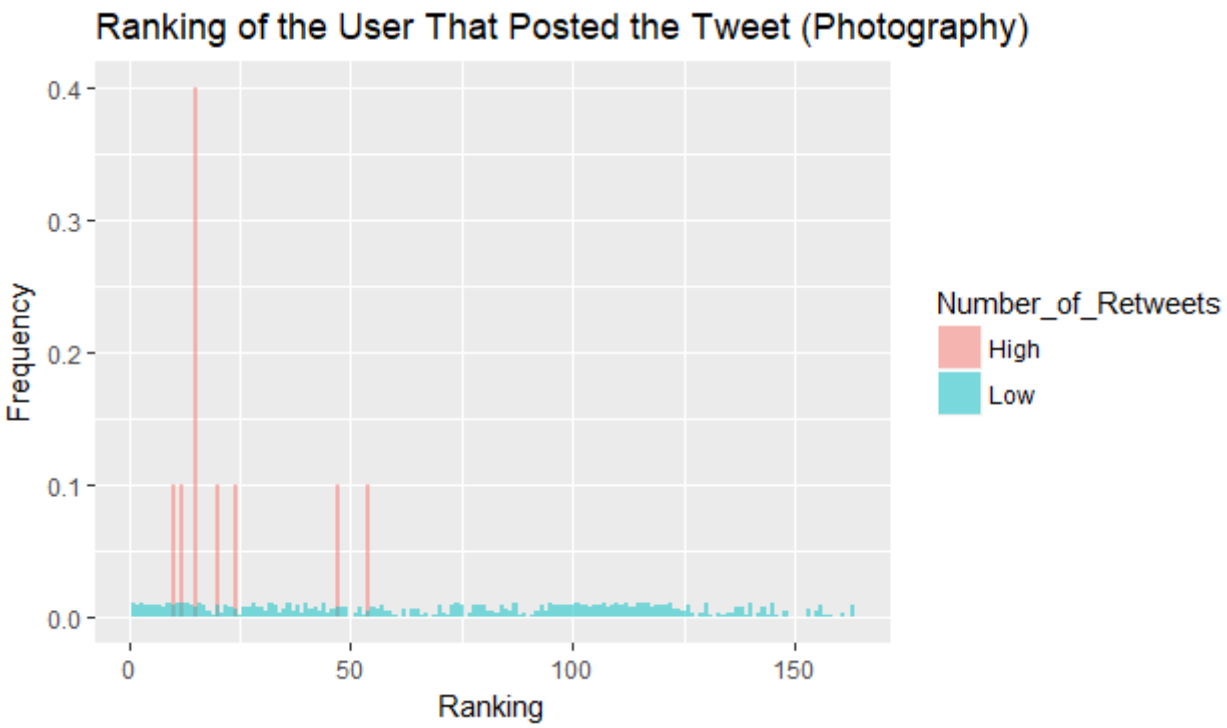

Figure B.4

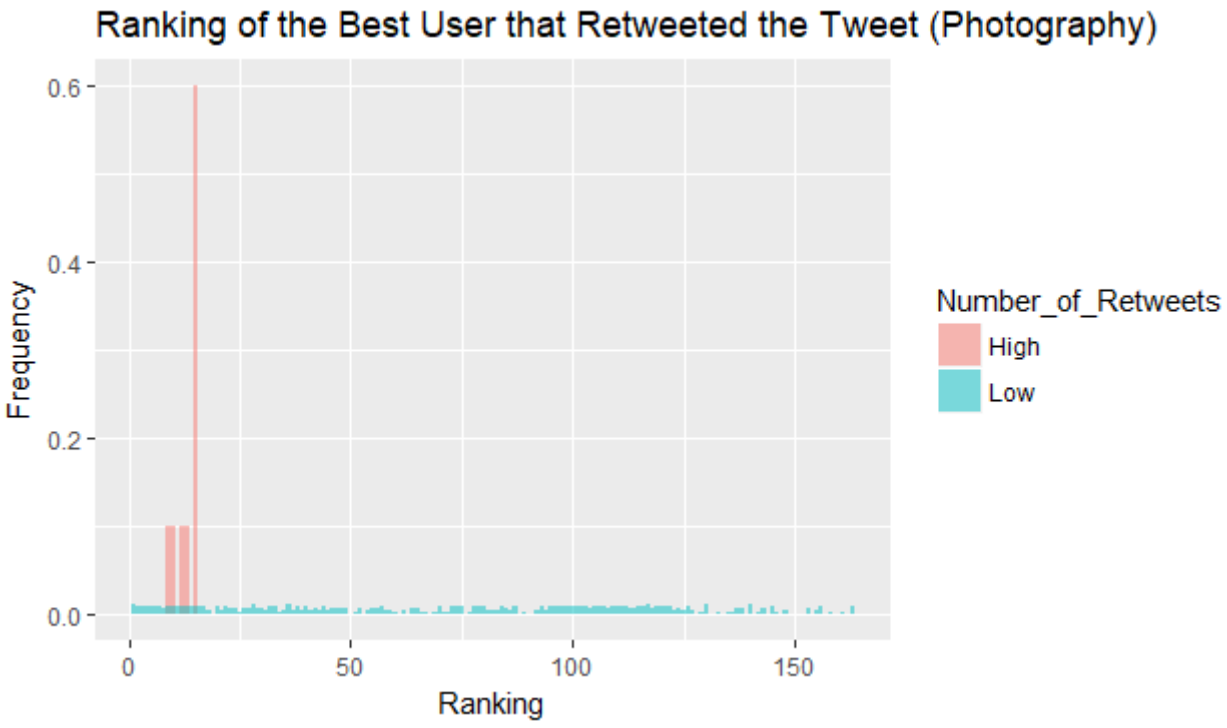

Figure B.5

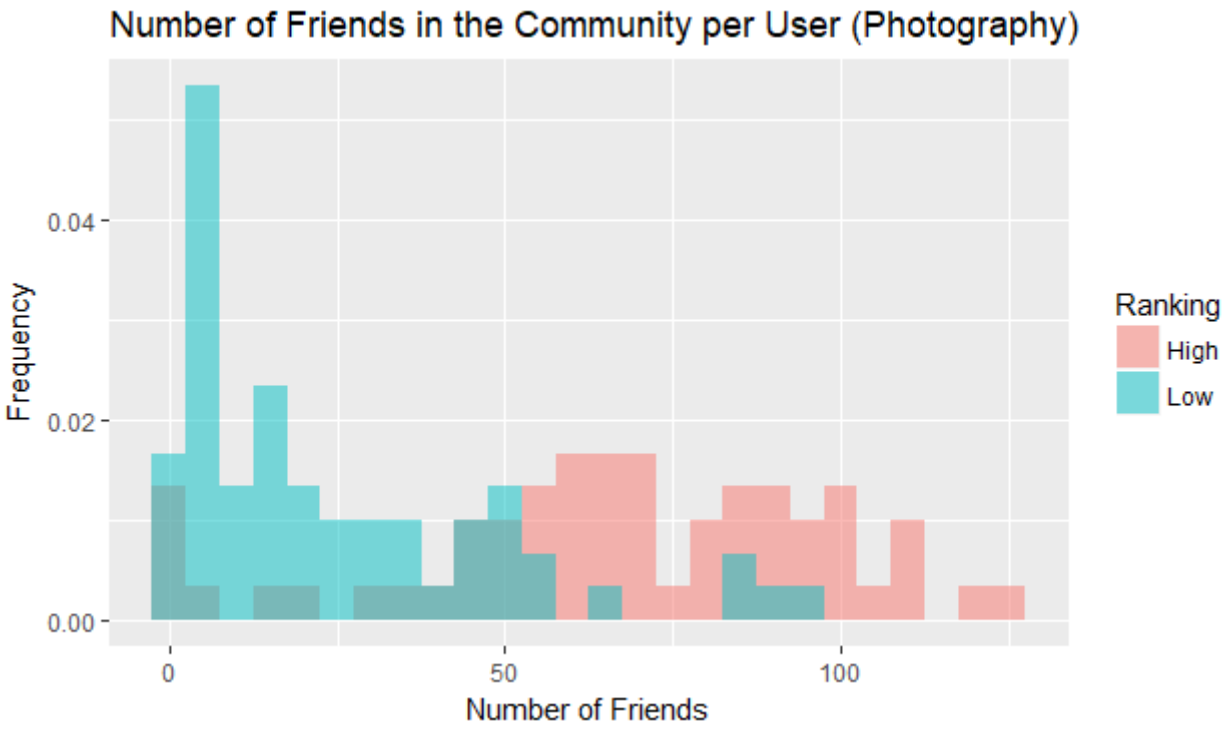
Number of Friends in the Community per User (Photography)
Frequency
0.04
0.02
0.00
0
50
100
Number of Friends
Ranking
High
Low

Figure B.6

Bibliography

[1] API reference index – Twitter Developers. https://developer.twitter.com/en/docs/api-reference-index.

[2] MongoDB Documentation. https://docs.mongodb.com/.

[3] The Guardian. Russia used hundreds of fake accounts to tweet about Brexit, data shows. Available online at https://www.theguardian.com/world/2017/nov/14/how-400-russia-run-fake-accounts-posted-bogus-brexit-tweets.

[4] Tweepy Documentation – tweepy 3.5.0 documentation. http://tweepy.readthedocs.io/en/v3.5.0/.

[5] H. Allcott and M. Gentzkow. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31(2):211–236, Spring 2017.

[6] A.-L. Barabási. *Network Science*, chapter 3. Available online at http://www.networksciencebook.com/.

[7] A. Sanromà. Study of Microscopic Properties of Information Communities in Online Social Networks. Bachelor's Degree Thesis. *Universitat Politècnica de Catalunya*, 2017.

[8] L. Zhang and P. Marbach. "Two is a Crowd" - Optimal Trend Adoption in Social Networks. *University of Toronto*, 2011.